Oracle® Fusion Applications
Performance and Tuning Guide
11g Release 1 (11.1.2)
E16686-01

October 2011
Documentation for administrators that describes how to monitor and optimize performance for Oracle Fusion Applications.
## Contents

### Preface
- Audience ................................................................. vii
- Documentation Accessibility ................................. vii
- Related Documents .................................................. vii
- Conventions .............................................................. viii

### 1 Monitoring and Tuning Oracle Fusion Applications
1.1 Introduction .......................................................... 1-1
1.1.1 Audience ........................................................ 1-1
1.2 Monitoring and Tuning Oracle Fusion Applications .... 1-1
1.2.1 How to Analyze Host Metrics ............................. 1-2
1.2.2 How to Check for Network Connectivity Issues ...... 1-4
1.2.3 How to Analyze WebLogic Server Metrics ............ 1-5
1.2.4 How to Analyze Oracle HTTP Server Metrics....... 1-6
1.2.5 How to Analyze Oracle Business Intelligence Server Metrics 1-7
1.2.6 How to Gather Key Identity Management Server Metrics 1-8
1.2.7 How to Analyze Key Enterprise Scheduler Metrics ... 1-8
1.2.8 How to Monitor Key SOA Metrics ..................... 1-9
1.2.9 How to Analyze Key Enterprise Scheduler Metrics ... 1-9
1.3 Tuning Platforms for Fusion Applications ................ 1-11

### 2 Troubleshooting Process
2.1 Introduction to Troubleshooting .......................... 2-1
2.1.1 How to Troubleshoot Overall System Slowness ...... 2-2
2.1.2 How to Find the Slowest JSPX/JSFF Pages .......... 2-3
2.1.3 How to Troubleshoot Slow UI Request ............... 2-3
  2.1.3.1 Troubleshooting Historical Requests .............. 2-4
  2.1.3.2 Troubleshooting Live Requests .................... 2-4
  2.1.3.3 Troubleshooting StuckThreads .................... 2-4
  2.1.3.4 Troubleshooting Slow Requests Using JFR Recording 2-6
  2.1.3.5 Troubleshooting Memory Leaks and Heap Usage Pressure 2-6
2.1.4 How to Troubleshoot Slow Batch Jobs ............... 2-7
  2.1.4.1 Troubleshooting Jobs That Are in Wait/Ready/Blocked State for a Long Time 2-7
  2.1.4.2 Troubleshooting Jobs in Running State for a Long Time 2-7
  2.1.4.3 Troubleshooting Slow BI Publisher Jobs .......... 2-8
3 Tuning the Database

3.1 Introduction ................................................................. 3-1
3.1.1 Monitoring the Oracle Fusion Applications Database .......... 3-2
3.2 Tuning the Database ..................................................... 3-2
3.2.1 How to Configure the Database for Performance .......... 3-4
3.2.2 How to Configure Kernel Parameters ......................... 3-5
3.2.3 How to Configure the Database Listener .................... 3-5
3.2.4 How to Tune the Real Application Cluster (RAC) ......... 3-6
3.2.5 How to Optimize SQL Statements .......................... 3-8
3.2.5.1 Collecting Optimizer Statistics .......................... 3-8
3.2.5.2 Pinning Packages and Cursors .......................... 3-10
3.2.6 How to Configure the Database Resource Manager .......... 3-10
3.2.6.1 Assigning Connections to Resource Consumer Groups .... 3-11
3.2.6.2 Using the Oracle Database Resource Manager ........... 3-11
3.2.6.3 Monitoring the Resource Manager ....................... 3-14
Preface

Welcome to the Oracle Fusion Applications Performance and Tuning Guide! This guide describes performance and tuning checks and tweaks that are specific to Oracle Fusion Applications.

Audience

This document is intended for Oracle Fusion Applications administrators and developers, and assumes familiarity with Java and SQL.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Access to Oracle Support

Oracle customers have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

Related Documents

For more information, see the following documents in the Oracle 11g Fusion Middleware documentation set:

- Oracle Fusion Applications Administrator's Guide
- Oracle Application Server Administrator's Guide
- Oracle JRockit JDK Tools Guide
- Oracle JRockit Flight Recorder Run Time Guide
- Oracle Fusion Middleware Managing Server Startup and Shutdown for Oracle WebLogic Server
- Oracle Fusion Middleware Configuring and Using the Diagnostics Framework for Oracle WebLogic Server
## Conventions

The following text conventions are used in this document:

<table>
<thead>
<tr>
<th>Convention</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>boldface</strong></td>
<td>Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.</td>
</tr>
<tr>
<td><em>italic</em></td>
<td>Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.</td>
</tr>
<tr>
<td><code>monospace</code></td>
<td>Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.</td>
</tr>
</tbody>
</table>
Monitoring and Tuning Oracle Fusion Applications

This chapter discusses how to find the information you need to examine so you can tune your system. It includes how to monitor and tune the database and Oracle Fusion Applications, and troubleshooting.

This chapter includes these sections:

■ Section 1.1, "Introduction"
■ Section 1.2, "Monitoring and Tuning Oracle Fusion Applications"
■ Section 1.3, "Tuning Platforms for Fusion Applications"

1.1 Introduction

Every system of hardware and installed applications is different. Even though Oracle Fusion Applications are written and installed using industry-standard best practices, you can custom tailor your system to improve how it supports your environment.

But to tune your system, you need to locate and examine data. This chapter will explain what data you need to examine, and what tools you will use to gather the data.

1.1.1 Audience

The Oracle Fusion Applications Performance Tuning Guide is intended for developers who are customizing an application, and operators working in a runtime production environment.

1.2 Monitoring and Tuning Oracle Fusion Applications

In general, most of the settings that come default in Oracle Fusion Applications are already tuned.

These guidelines are provided to help ensure your Oracle Fusion Applications instance runs optimally. Note that all metrics listed are from Oracle Enterprise Manager Cloud Control.

■ Monitor the key host metrics, shown in Table 1–1, to ensure the underlying server hosts are healthy. Rather than constantly checking the metric values, you can set up alert thresholds in Cloud Control and receive notification when thresholds are exceeded.

■ Monitor the key component metrics, such as WebLogic server metrics, to ensure each component is healthy.
- Monitor the number of incidents and logs to ensure the application is configured properly and not constantly wasting resources generating error messages. Review log levels to ensure they are not set too low. See “Troubleshooting Oracle Fusion Applications Using Incidents, Logs, QuickTrace, and Diagnostic Tests” in the Oracle Fusion Applications Administrator’s Guide for more information.

- Monitor the database to ensure it is operating optimally. Follow the guidelines in Chapter 3, "Tuning the Database," to make sure that statistics are being collected.

**Table 1–1 Key Host Metrics**

<table>
<thead>
<tr>
<th>Metric Category</th>
<th>Metric Name</th>
<th>Warning Threshold</th>
<th>Critical Threshold</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Disk Activity</td>
<td>Disk Device Busy</td>
<td>&gt;80%</td>
<td>&gt;95%</td>
<td></td>
</tr>
<tr>
<td>Filesystems</td>
<td>Filesystem Space Available</td>
<td>&lt;20%</td>
<td>&lt;5%</td>
<td></td>
</tr>
<tr>
<td>Load</td>
<td>CPU in I/O wait</td>
<td>&gt;60%</td>
<td>&gt;80%</td>
<td></td>
</tr>
<tr>
<td></td>
<td>CPU Utilization</td>
<td>&gt;80%</td>
<td>&gt;95%</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Run Queue (5 min average)</td>
<td>&gt;2</td>
<td>&gt;4</td>
<td>The run queue is normalized by the number of CPU cores.</td>
</tr>
<tr>
<td></td>
<td>Swap Utilization</td>
<td>&gt;75%</td>
<td>&gt;90%</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Total Processes</td>
<td>&gt;15000</td>
<td>&gt;25000</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Logical Free Memory %</td>
<td>&lt;20</td>
<td>&lt;10</td>
<td></td>
</tr>
<tr>
<td></td>
<td>CPU in System Mode</td>
<td>&gt;20%</td>
<td>&gt;40%</td>
<td></td>
</tr>
<tr>
<td>Network Interfaces</td>
<td>All Network Interfaces Combined Utilization</td>
<td>&gt;80%</td>
<td>&gt;95%</td>
<td></td>
</tr>
<tr>
<td>Summary</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Switch/Swap Activity</td>
<td>Total System Swaps</td>
<td>&gt;3</td>
<td>&gt;5</td>
<td>Value is per second.</td>
</tr>
<tr>
<td>Paging Activity</td>
<td>Pages Paged-in (per second)</td>
<td></td>
<td></td>
<td>The combined value of Pages Paged-in and Pages Paged-out should be &lt;=1000</td>
</tr>
<tr>
<td></td>
<td>Pages Paged-out (per second)</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**1.2.1 How to Analyze Host Metrics**

Administrators will find it useful to study these suggestions on further analysis to undertake when a metric value exceeds threshold. The commands provided are for the Linux operating system.

**When logical free memory/swap activity or paging activity is beyond threshold**

This usually happens when memory is not sufficient to handle demands from all the running processes.

- Check `cat/proc/meminfo` and confirm total RAM is expected.

- Check if there are unallocated huge pages. If there are and the WebLogic Server/Oracle instances are not expected to use them, reduce the huge page pool size.

- Run `top` and sort by resident memory (type `OQ`). Look for processes using the most resident memory and investigate those processes.
When page activity is beyond threshold
Follow the steps in "When logical free memory/swap activity or paging activity is beyond threshold" to view and analyze memory usage.

When Network Interface Error Rates Is Beyond Threshold
The normal cause is misconfiguration between the host and the network switch. A bad network card or cabling also can cause this error. You can run /sbin/ifconfig to identify which interface is having packet errors. Contact network administrator to ensure the host and the switch are using same data rate and duplex mode.

Otherwise, check if cabling or the network card is faulty and replace as appropriate.

When Packet Loss Rate Is Beyond Threshold
The normal cause of this error is network saturation of bad network hardware.

- Run lsof -Pni | grep ESTAM to determine which network paths are generating the problem.
- Then run mtr <target host> or ping <target host> and look for packet lost on that segment.

```
20 packets transmitted, 20 received, 0% packet loss, time 18997ms
  rtt min/avg/max/mdev = 0.168/0.177/0.200/0.010 ms
```

The packet loss should be 0% and rtt should be less than .5 ms.

- Ask the network monitoring staff to look for saturation or network packet loss from their side.

When Network Utilization Is Beyond Threshold
The normal cause is very heavy application load.

- Run top or lsof to determine which processes are moving a lot of data.
- Use tcpdump to sample the network for usage patterns.
- Use atop, iftop, ntop or pkstat to see which processes are moving data.

When CPU Usage or Run Queue Length Is Beyond Threshold
The normal cause is runaway demand, a poorly performing application, or poor capacity planning.

- Run top to identify which application/process is using time.
- If top processes are WebLogic Server JVM processes, conduct a basic WebLogic Server health check. That is, review logs to see if there are configuration errors causing excessive exceptions, and review metrics to see if the load has increased. Use JVMD for a more detailed analysis.
- If top processes are Oracle processes, use Enterprise Manager to look for high load SQL.

When System CPU Usage Is Beyond Threshold

- High system CPU use could be due to kernel processes looking for pages to swap out during a memory shortage. Follow the steps listed in "When logical free memory/swap activity or paging activity is beyond threshold" section to further diagnose the problem.
- High system CPU use is also frequently related to various device failures. Run 

  ```
  (dmesg | less)
  ```

  and look for repeated messages about errors on some
particular device, and also have hardware support personnel check the hardware console to see if there are any errors reported.

**When Filesystem Usage Is Beyond Threshold**
The normal cause is an application that is logging excessively or leaving behind temporary files.

- Run `lsf -d 1-99999 | grep REG | sort -nrk 7 | less` to see currently open files sorted by size from largest to smallest. Investigate the large files.
- Run `du -k /mount_point_running_out_of_space | tmp/sizes` to get space used for directories under the mount point. This may take a long time. While it is running, run `sort -nr /tmp/sizes` and find the directories using most space and investigate those first.

**When Total Processes Is Beyond Threshold**
The normal cause is runaway code or a stuck NFS filesystem.

- Run `ps aux`. If many processes are in status D, run `df` to check for stuck mounts.
  If there are hundreds or thousands of processes of a particular program, determine why.
- Run `ps o pid,nlwp,cmd | sort -nrk 2 | head` to look for processes with many threads.

**When Disk Device Busy Is Beyond Threshold**
- Check for disk drive failure. As root, check `/var/log/messages*` and `/var/log/mcelog` to see if there are any error messages indicating disk failure. For a RAID array, the disk controller needs to be checked. The commands will be specific to the controller manufacturer.
- Look for processes that are using the disk. From a shell window, execute `ps aux | grep 'D.'` several consecutive times to look for processes with "stat" D.

### 1.2.2 How to Check for Network Connectivity Issues
Poor performance is a major indicator of network connectivity problems.

- Check for cumulative dropped packets drops for each host.
  ```
  netstat -s | grep 'TCP data loss'
  4007 segments retransmitted
  3302 TCP data loss events
  ```
  These counts should be 0 or growing very slowly over time.

- Check for realtime dropped packets on specific network paths.
  ```
  ping -c 20 other_host
  20 packets transmitted, 20 received, 0% packet loss, time 18997ms
  rtt min/avg/max/mdev = 0.168/0.177/0.200/0.010 ms
  ```
  Packet loss should be 0%.
  
rtt should be less than .5 ms, except that it can be higher between the browser and load balancer.

- Check for network interface errors.
1.2.3 How to Analyze WebLogic Server Metrics

These metrics provide an indication of whether the WebLogic Server is in a healthy state. Performance may degrade if any of the metrics is exceeding its threshold.

See "Monitoring the Oracle Fusion Applications Middle Tier" in the Oracle Fusion Applications Administrator's Guide and Table 1–2.

**Table 1–2  Key WebLogic Server Metrics**

<table>
<thead>
<tr>
<th>Metric Category</th>
<th>Metric Name</th>
<th>Warning Threshold</th>
<th>Critical Threshold</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Datasource Metrics</td>
<td>Connections in Use</td>
<td>&gt;250</td>
<td>&gt;400</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Connection Requests that Waited (%)</td>
<td>&gt;10%</td>
<td>&gt;20%</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Connection Creation Time (ms)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>JVM Garbage Collectors</td>
<td>Garbage Collector - Percent Time spent (elapsed)</td>
<td>&gt;10%</td>
<td>&gt;20%</td>
<td></td>
</tr>
<tr>
<td>JVM Metrics</td>
<td>Heap Usage</td>
<td>&gt;90%</td>
<td>&gt;98%</td>
<td></td>
</tr>
<tr>
<td>Response</td>
<td>Status</td>
<td></td>
<td>=Down</td>
<td>This provides instance availability.</td>
</tr>
<tr>
<td>Server Servlet/JSP Metrics</td>
<td>Request Processing Time (ms)</td>
<td>&gt;10s</td>
<td>&gt;15s</td>
<td></td>
</tr>
<tr>
<td>Server Work Manager Metrics</td>
<td>Work Manager Stuck Threads</td>
<td>&gt;5</td>
<td>&gt;10</td>
<td></td>
</tr>
<tr>
<td>JVM Threads</td>
<td>Deadlocked Threads</td>
<td>&gt;2</td>
<td>&gt;5</td>
<td></td>
</tr>
<tr>
<td>Module Metrics By Server</td>
<td>Active Sessions</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

When CPU Usage On Host Is Beyond Threshold and WebLogic Server Process Is Identified as Top CPU Consumer

- Examine the % Time spent in the GC metric to see if JVM is doing excessive GC (>60 percent). If so, follow the process for diagnosing WebLogic Server heap pressure.
- Look for incident creation rate and error logs and see if something is triggering a massive amount of logging/errors.
- In JVMD, select the CPU state filter and look at top methods. Look for threads that are consistently in a CPU state.

When There Is a Spike in Active Web Sessions

- Check access logs to see if there is a spike in the number of users.
- Check if there are stuck threads, which could cause users to log in again.
- Check session distribution across WebLogic Server managed servers and see if there is a problem with the load balancer.
- Check session timeout in **web.xml**, and see if it is too high or too low.
When There Are Stuck Threads On the System

■ Get the ECID from the stuck thread error in the WebLogic Server log.
■ From the Request Monitor, search for the ECID and get details from JVMD.
■ Alternatively, use JVMD to search for stuck threads and see the timing breakdown.
■ A stuck thread will also result in an incident with a JFR recording. Use JRMC to analyze the recording.

When There Are Deadlocks Detected On the System

■ In JVMD, inspect the threads that are in a blocked state.
■ Deadlock threads normally also will be reported as a stuck thread in the WebLogic Server log. Use the Request Monitor to search for the ECID and expand down into JVMD to show the blocking thread.

When Request Processing Time Is Beyond Threshold

■ Examine the % Time spent in GC metric to see if JVM is doing excessive garbage collection
■ Look for incident create rate and error logs and see if something is triggering a massive amount of logging/errors.
■ In JVMD, look at the thread states and see where most processing time is going.

When Percent Time Spent in GC Is Beyond Threshold

■ Check the session count. If there is a sudden surge of sessions due to user load, the JVM could be short on heap. Increase heap if possible, or add additional managed server instances.
■ Look at the stuck threads count. Stuck threads could increase the number of active session, as users could be launching new sessions hoping for a faster response.
■ Look at the incident creation rate and error logs and see if something is triggering a massive amount of logging/errors. The incident creation/logging operations could be causing a high amount of object creation and garbage collection stress.
■ Generate a heap dump using JVMD and analyze the top retainer of memory.
■ Use JRMC to connect and extract a JFR recording. Examine the Memory panel and allocation details to see what is doing a lot of allocations.

When Percent Connection Requests Waiting Is Beyond Threshold

■ Examine the number of sessions and request rate, and see if there is a spike in the load that would account for an increased demand for connections.
■ In JVMD, see where time is spent. For example, requests could be running longer due to slow SQLs (and retain the connection longer). In that case, identify and tune slow SQLs.
■ Consider increasing the initial capacity setting of the corresponding data source.

1.2.4 How to Analyze Oracle HTTP Server Metrics

These metrics provide an indication of whether the Oracle HTTP Server is in a healthy state. Performance may degrade if any of the metrics is exceeding its threshold.
See "Monitoring the Oracle Fusion Applications Middle Tier" in the Oracle Fusion Applications Administrator’s Guide and Table 1–3.

**Table 1–3  Oracle HTTP Server Metrics**

<table>
<thead>
<tr>
<th>Metric Category</th>
<th>Metric Name</th>
<th>Warning Threshold</th>
<th>Critical Threshold</th>
</tr>
</thead>
<tbody>
<tr>
<td>OHS Server Metrics</td>
<td>Busy Threads (%)</td>
<td>&gt;85%</td>
<td>&gt;95%</td>
</tr>
<tr>
<td></td>
<td>Request Throughput (requests per second)</td>
<td>TBD</td>
<td>Yes</td>
</tr>
<tr>
<td>OHS Response Code Metrics</td>
<td>HTTP 4xx errors</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>HTTP 5xx errors</td>
<td></td>
<td></td>
</tr>
<tr>
<td>OHS Virtual Host Metrics</td>
<td>Request Processing Time for a Virtual Host</td>
<td>&gt;10s</td>
<td>&gt;15s</td>
</tr>
</tbody>
</table>

---

When Busy Threads % Is Beyond Threshold

- Check request throughput to see if load has increased. If the increased load is expected and CPU and memory resources on the OHS host has not exceeded threshold, consider increasing ServerLimit/MaxClients and ThreadsPerChild in `httpd.conf`.
- Check request process time on both OHS and underlying WebLogic Server to see if requests are taking longer. If WebLogic Server response time is increasing, check the key metrics for the WebLogic Server.
- If possible, ensure the client browser cache is enabled to reduce number of requests submitted.
- Check OHS Response Code Metrics. If there is a sudden increase of HTTP 4xx errors or HTTP 5xx errors, check the health of the underlying WebLogic Servers.
- Check and increase the minimum and maximum spare threads for Oracle HTTP Server.

In the `httpd.conf` file located in `instance_home/config/ohs/<ohs_name>/httpd.conf`:

- Increase MaxSpareThreads to 800.
- Increase MinSpareThreads to 200.

When Request Processing Time for a Virtual Host Exceeds Threshold

- Check the key host metrics to ensure the OHS host is healthy.
- For each URL requested, OHS will first check DocumentRoot before passing the request to WebLogic Server. Check the utilization and health of the disk to which the DocumentRoot is pointing. If it is a NFS mount, check the health of the NFS mount point.
- Check the key metrics for the underlying WebLogic Server(s) and see if they are healthy.
- OHS accesses /tmp for each POST request, so check the performance of the /tmp filesystem.

1.2.5 How to Analyze Oracle Business Intelligence Server Metrics

These metrics provide an indication of whether the Oracle Business Intelligence Server is in a healthy state.
To start monitoring:

- Log in to Oracle Enterprise Manager Fusion Applications Control.
- Open Business Intelligence > coreapplication > Business Intelligence Instance > Monitoring > Performance, as shown in Figure 1–1.

**Figure 1–1 Checking Oracle Business Intelligence Performance Metrics**

- Use Fusion Applications Control to configure parameters related to Oracle Business Intelligence Suite Enterprise Edition.

Fusion Applications Control can monitor various BI components, including:

- Weblogic Analytics Application
- Oracle BI Presentation Services
- Oracle BI Server
- Oracle Weblogic Server (administration and managed servers)

### 1.2.6 How to Gather Key Identity Management Server Metrics

Oracle Access Manager and Oracle Identity Manager are both WebLogic Server instances. See Section 1.2.3, "How to Analyze WebLogic Server Metrics" to monitor their health.

Use Cloud Control to monitor the Oracle Internet Directory and Oracle Identity Manager databases.

### 1.2.7 How to Analyze Key Enterprise Scheduler Metrics

These metrics provide an indication of whether the Enterprise Scheduler instance is performing well.

**Table 1–4 Key Enterprise Scheduler Metrics**

<table>
<thead>
<tr>
<th>Metric Category</th>
<th>Metric Name</th>
<th>Warning Threshold</th>
<th>Critical Threshold</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Completed Job Summary</td>
<td>Average Elapsed Time (ms)</td>
<td></td>
<td></td>
<td>You can define different thresholds for different job names.</td>
</tr>
<tr>
<td>Long Running Job</td>
<td>Elapsed Time (ms)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>WorkAssignment Metrics aggregated across Group Members</td>
<td>Average Wait Time for Requests in Ready State (seconds)</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
When the Value of Average Elapsed Time for the Completed Jobs Is Higher Than Expected

- Check the key host/WebLogic Server metrics and see if any component that could be involved in process batch jobs is in an unhealthy state.
- Follow the steps listed in the "Troubleshooting Slow Batch Job" section and analyze a few jobs to see if there are any common causes.

When the Value of Elapsed Time Under the Long Running Job Category Is Higher Than Expected

- Open the Enterprise Scheduler home page in Oracle Enterprise Manager Fusion Applications Control and examine the Top 10 Long Running Jobs.
- Identify the job of interest, and follow the steps in the "Troubleshooting Slow Batch Job" section.

When Average Wait Time For Requests in Ready State (seconds) Is Higher Than Expected

Follow the steps in "Troubleshooting Jobs that are in Wait/Ready/Blocked state for a long time" section.

1.2.8 How to Monitor Key SOA Metrics

Monitoring SOA involves monitoring SOA infrastructure, SOA composite and SOA servers.

See "Monitoring the Oracle Fusion Applications Middle Tier" in the Oracle Fusion Applications Administrator’s Guide.

1.2.9 How to Tune and Troubleshoot Oracle Identity Management

Follow the steps in this section to tune Oracle Identity Management specifically for Oracle Fusion Applications.

Optimize LDAP Search

Description: Optimize LDAP search by enabling search filters.

Solution:

- Create a ldiff file named searchfilter_oid_tuning.ldif with this content:

```diff
  dn: cn=dsacfg, cn=configsets, cn=oracle internet directory
  changeType: modify
  add: orclinmemfiltprocess;dn
  orclinmemfiltprocess;dn:
  cn=Roles,cn=fscm,cn=FusionDomain,cn=JPContext,cn=FusionAppsPolicies
  orclinmemfiltprocess;dn:
  cn=Roles,cn=crm,cn=FusionDomain,cn=JPContext,cn=FusionAppsPolicies
  orclinmemfiltprocess;dn:
  cn=Roles,cn=hcm,cn=FusionDomain,cn=JPContext,cn=FusionAppsPolicies
  orclinmemfiltprocess;dn:
  cn=Permission
  Sets,cn=fscm,cn=FusionDomain,cn=JPContext,cn=FusionAppsPolicies
  orclinmemfiltprocess;dn:
  cn=Permission
  Sets,cn=hcm,cn=FusionDomain,cn=JPContext,cn=FusionAppsPolicies
  orclinmemfiltprocess;dn:
  cn=Permission
  Sets,cn=crm,cn=FusionDomain,cn=JPContext,cn=FusionAppsPolicies
  orclinmemfiltprocess;dn:
  cn=Permissions,cn=JAAS
  Policy,cn=fscm,cn=FusionDomain,cn=JPContext,cn=FusionAppsPolicies
  orclinmemfiltprocess;dn:
```

```- MODIFY
  dn: orclinmemfiltprocess;dn
```
At the command prompt, run this command:

```
ldapmodify -p portNum -h hostname -D cn=orcladmin -f searchfilter_oid_tuning.ldif
```

**Log Levels**

**Description:** Oracle Identity Management stack WebLogic Server log levels are too fine-grained and need to be set to Severe.

**Solution:** In all WebLogic Servers in the Oracle Identity Management domain, change log levels to SEVERE. This is a two-part process.

- **Part 1:** Manually edit the `logging.xml` file, or by using the Oracle WebLogic Server Administration Console.

  Edit the `logging.xml` file that is in each server directory of the Oracle Identity Management Domain domain, such as OAM_Server1, OIM_Server1, and SOA, and set `level='SEVERE'` for all log_handlers and loggers. The path to each `logging.xml` file will resemble:

  `$domain_home/config/fmwconfig/<servername>`

- **Part 2:** Edit the log levels in the Oracle WebLogic Server Administration Console:
  - Log in to the console (http://hostname:port/console).
  - Click the Servers link.
  - Click the desired server.
  - Click the **Logging** tab.
  - Scroll down and click the **Advanced** link.
  - In the Message destination(s) section, change the log levels as shown here:

    ```
    Log file : Severity level: warning
    Standard out :Severity level: error
    Domain log broadcaster :Severity level: error
    Memory buffer: Memory Buffer Severity level: error
    ```

    - Save the changes.
    - Repeat this for all WebLogic Servers in the Oracle Identity Management stack, such as OAM_Server1, OIM_Server1, and SOA.
    - Click **Activate Changes**.
    - Restart the server.

**Avoid Restarts of httpd-worker Processes**

**Description:** These restarts affect the recreation of connections and threads in Oracle HTTP Server processes during varying load patterns.

**Solution:** Increase the minimum and maximum spare threads for Oracle HTTP Server.

In the `httpd.conf` file located in `instance_home/config/ohs/<ohs_name>/httpd.conf:`
- Increase MaxSpareThreads to 800.
- Increase MinSpareThreads to 200.

**Tune Two OID Configuration Parameters**

**Description:** Two OID configuration parameters, orclmaxcc and orclserverprocs, need to be appropriately tuned.

**Solution:** Change `orclmaxcc` to 10 and tune the number of OID processes:

- Name the sample script `config_oid_tuning.ldif`. You will need to set `cn=oid1` to your component name. In a multi-component environment, this needs to be changed accordingly. You will need to set orclserverprocs to the number of cores in the OID server that is used.

```
dn: cn=oid1,cn=osdldapd,cn=subconfigsubentry
changetype: modify
replace: orclmaxcc
orclmaxcc: 10
orclserverprocs: <number of cores>
```

- Apply the script by running this command at the command prompt:

```
ldapmodify -p portNum -h hostname -D cn=orcladmin -f config_oid_tuning.ldif
```

**Enable Timing Logging**

**Description:** Add parameters to enable timing logging for OID.

**Solution:**

- Add this entry to the config.xml file in `/oid/user_projects/domains/oid_domain/config/` and the `/oim/user_projects/domains/oim_domain/config/` directories for each WebLogic Server in the Oracle Identity Management domain:

```
<web-server>
    <web-server-log>
        <file-name>logs/access.log.%yyyyMMdd%</file-name>
        <rotation-type>byTime</rotation-type>
        <number-of-files-limited>true</number-of-files-limited>
        <rotate-log-on-startup>true</rotate-log-on-startup>
        <buffer-size-kb>0</buffer-size-kb>
        <logging-enabled>true</logging-enabled>
        <elf-fields>date time time-taken bytes c-ip s-ip sc-status sc(X-ORACLE-DMS-ECID) cs-method cs-uri cs(User-Agent) cs(ECID-Context) cs(Proxy-Remote-User) cs(Proxy-Client-IP)</elf-fields>
        <log-file-format>extended</log-file-format>
        <log-time-in-gmt>false</log-time-in-gmt>
        <log-milli-seconds>true</log-milli-seconds>
    </web-server-log>
</web-server>
```

- To set the access log format, add this string to the `httpd.conf` file in the `/u01/ohsauth/ohsauth_inst/config/OHS/ohs1` path.

```
LogFormat "%h %l %u %t \"%r\" %>s %b %D %{X-ORACLE-DMS-ECID}o" common
```

### 1.3 Tuning Platforms for Fusion Applications

If you are using the Solaris SPARC or the IBM AIX operating system, Oracle recommends that you incorporate these settings.
Solaris SPARC
Incorporate this setting for best performance when using Hotspot JVM:

-XX:+UseParallelOldGC  -XX:ParallelGCThreads=4

IBM AIX
Incorporate this setting for best performance when using IBM JVM 9:

XgcPolicy:gencon -Xcompressedrefs -XtlhPrefetch
This chapter discusses the basic process to use to troubleshoot Oracle Fusion Applications, and presents specific steps for the most commonly-seen problems.

This chapter includes these sections:

- Section 2.1, "Introduction to Troubleshooting"
- Section 2.2, "View Detailed Timing Of a Request Using a JRockit Flight Recorder (JFR) File"
- Section 2.3, "Using My Oracle Support for Additional Troubleshooting Information"

2.1 Introduction to Troubleshooting

This section provides guidelines and a process for using the information in this chapter. Using the following guidelines and process will focus and minimize the time you spend resolving problems.

Guidelines

When using the information in this chapter, Oracle recommends:

- After performing any of the solution procedures in this chapter, immediately retrying the failed task that led you to this troubleshooting information. If the task still fails when you retry it, perform a different solution procedure in this chapter and then try the failed task again. Repeat this process until you resolve the problem.

- Making notes about the solution procedures you perform, symptoms you see, and data you collect while troubleshooting. If you cannot resolve the problem using the information in this chapter and you must log a service request, the notes you make will expedite the process of solving the problem.

Process

Follow the process outlined in Table 2–1 when using the information in this guide. If the information in a particular section does not resolve your problem, proceed to the next step in this process.
In addition to this process, for more information about determining if database cache sizes need to be increased, see “Automatic Database Performance Monitoring” chapter in the Oracle Database 2 Day + Performance Tuning Guide to use Automatic Database Diagnostic Monitor (ADDM) reports.

### Table 2–1 Process for Resolving Performance Issues

<table>
<thead>
<tr>
<th>Step</th>
<th>Section to Use</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| 1    | Chapter 1, "Monitoring and Tuning Oracle Fusion Applications" | Collect symptoms about the performance problem to determine if the problem is related the following:  
- Response time or throughput  
- Widespread or limited to specific users and flows  
Oracle Fusion Applications Administrator's Guide:  
Chapter 9 and Chapter 10 for locating key metrics  
Chapter 11 for diagnosing Java applications in the middle tier  
Chapter 12 for monitoring and tuning the Oracle database  
Determine what changed since the system was last performing well. |
| 2    | Section 2.1.1 through Section 2.1.6 | Use Section 2.1.1 if the problem is widespread. Otherwise, review the problem description in Section 2.1.2 through Section 2.1.6 to if there is a match  
These sections describe:  
- Possible causes of the problems  
- Solution procedures corresponding to each of the possible causes |
| 4    | Section 2.3 | Use My Oracle Support to get additional troubleshooting information about Oracle Fusion Applications or performance, scalability, and reliability. My Oracle Support provides access to several useful troubleshooting resources, including Knowledge Base articles and Community Forums and Discussions. |
| 5    | Section 2.3 | Log a service request if the information in this chapter and My Oracle Support does not resolve your problem. You can log a service request using My Oracle Support at https://support.oracle.com. |

In addition to this process, for more information about determining if database cache sizes need to be increased, see “Automatic Database Performance Monitoring” chapter in the Oracle Database 2 Day + Performance Tuning Guide to use Automatic Database Diagnostic Monitor (ADDM) reports.

### 2.1.1 How to Troubleshoot Overall System Slowness

To troubleshoot overall system slowness:

- Navigate to the Fusion Applications target in Enterprise Manager.
- Select the application that is having the problem, such as Payables.
- Navigate to the product home page, such as the Payables product home page.
- Select **Monitoring > System Performance** to open the System Performance page.
Click the name of the target metric name for Request Processing Time. A popup displays.

Click **Problem Analysis** to see the related metrics in the problem analysis page. You should be able to see the Source Metric and Related Metrics in the Problem Analysis page.

Check if any key metrics are beyond threshold, or look for sudden changes. In particular, check the **heap usage** metrics.

If the heap is constantly close to 100 percent, search using the string OutOfMemoryErrors in the Oracle WebLogic Server server_name.out in the following directories:

(UNIX)  `DOMAIN_HOME/servers/server_name/logs`

(Windows)  `DOMAIN_HOME\servers\server_name\logs`

If there are OutOfMemoryErrors, a heap dump would have been generated in the directory specified by the `-XX:HeapDumpPath` parameter from the Oracle WebLogic Server startup JVM option. Submit the heap dump to Oracle Support for further analysis of what is retaining memory. In many cases a bounce is needed to resolve the issue.

Scroll down and you will be able to see the Related Targets at the bottom of the page.

Select the Related Targets Topology to view the Topology of the target on which the metric is being analyzed.

Review incidents and logs by clicking the **View Related Log Messages** link at the top right side of the page. The target context, the chart time duration and related targets are chosen to do log search.

Follow the appropriate triage process depending on which metric is in question.

### 2.1.2 How to Find the Slowest JSPX/JSFF Pages

An administrator would want this information to understand which part of the application has poor performance.

To find the slowest JSPX/JSFF pages:

- Navigate to **Middleware Targets** in Oracle Enterprise Manager Fusion Applications Control.
- Select **Application Dependency and Performance** from the drop-down menu.
- Expand the ADF node in the tree on the left hand side of the screen.
- Click JSF pages and sort by response time.
- Look for the slowest page name, and find that page name in the tree on the left. Click to drill down.
- If applicable, it will show the managed beans for that page.

### 2.1.3 How to Troubleshoot Slow UI Request

A sluggish UI response directly affects each user’s experience. Administrators will want to use the suggested troubleshooting methods in this section to track down and fix slow UI response.
2.1.3.1 Troubleshooting Historical Requests
These steps are applicable to requests that have completed.

- Find Execution Context Identifiers (ECIDs) submitted by user and find the slowest requests using server logs. Compare response time as recorded in Oracle HTML Server (OHS) and WebLogic Server access logs. If there is a big gap, check OHS health. Otherwise, look at request details in WebLogic Server.

  Find the nodes that serviced the request by searching for the ECID using Request Monitor in Enterprise Manager.

  Drill down into the JVM Diagnostics screen.

  The data displayed should be in the context of that ECID. Observe the thread state transitions and check the call stacks to find the top methods and top SQL statements.

- Check if slowness is caused by customized code.

- If slowness is caused by lock contention, check the thread that is holding the lock and what it is doing.

- If the request is slow enough to cause a STUCK thread, an incident should have been generated. Search the log file for the incident number The incident directory would have a WebLogic Server diagnostic image which includes a JRockit Flight Recorder (JFR) recording from running WebLogic Server.

  If there are not enough details in the JVM Diagnostic data, follow instructions in Section 2.2, “View Detailed Timing Of a Request Using a JRockit Flight Recorder (JFR) File” to extract and view JFR recordings.

2.1.3.2 Troubleshooting Live Requests
To troubleshoot slow UI requests that are still running, extract and view a JFR recording following instructions in Section 2.2, “View Detailed Timing Of a Request Using a JRockit Flight Recorder (JFR) File” against the server on which the user session is running.

2.1.3.3 Troubleshooting StuckThreads

Problem
Stuck threads may result if the server is nearing out of memory. If the server is close to out of memory, all requests should slow down. To resolve an out-of-memory issue, see Section 2.2, “View Detailed Timing Of a Request Using a JRockit Flight Recorder (JFR) File.”

If a request is taking longer than 10 minutes, the stuck thread is reported to Oracle WebLogic Server server_name.out in the following directories:

(UNIX)  
DOMAIN_HOME/servers/server_name/logs

(Windows)  
DOMAIN_HOME\servers\server_name\logs

For example:

<Mar 4, 2011 7:44:08 AM PST> <Error> <WebLogicServer> <BEA-000337> <[STUCK] ExecuteThread: '19' for queue: 'weblogic.kernel.Default (self-tuning)' has been busy for "600" seconds working on the request  
"weblogic.servlet.internal.ServletRequestImpl@18986012[GET /productManagement/faces/PimDashboardUiShellPage?_afrLoop=1398820150000&_afrWindowMode=0&_adf.ctrl-state=a44e7uxcc_13 HTTP/1.1  
Accept: image/gif, image/x-xbitmap, image/jpeg, image/pjpeg,
application/x-shockwave-flash, application/x-ms-application, application/x-ms-x-bap, application/vnd.ms-excel, application/vnd.ms-powerpoint, application/msword, */*
Accept-Language: fr
UA-CPU: x86
...
], which is more than the configured time (StuckThreadMaxTime) of "600" seconds.
Stack trace:
{self-tuning}'' <alive, in native, suspended, priority=1, DARMON> {
jrockit.net.SocketNativeIO.readBytesPinned(SocketNativeIO.java: ???)
jrockit.net.SocketNativeIO.socketRead(SocketNativeIO.java: 24)
java.net.SocketInputStream.socketRead0(SocketInputStream.java: ???)
java.net.SocketInputStream.read(SocketInputStream.java: 107)
...

In this example, the request has been running longer than the configured 600 seconds.
Here is the associated stack trace showing the thread is stuck:
{self-tuning}'' <alive, in native, suspended, priority=1, DARMON> {
jrockit.net.SocketNativeIO.readBytesPinned(SocketNativeIO.java: ???)
jrockit.net.SocketNativeIO.socketRead(SocketNativeIO.java: 24)
java.net.SocketInputStream.socketRead0(SocketInputStream.java: ???)
java.net.SocketInputStream.read(SocketInputStream.java: 107)
...

Solution
If the stack shows the thread is waiting for a response from another server, check the
status of the other server and see it has performance problems before proceeding with
this solution.
To determine what the stuck thread was doing prior to becoming stuck, perform the
following steps:
1. Look at the next few log messages in server_name.out for a message indicating
   an incident has been created. For example:
   <Mar 4, 2011 7:44:10 AM PST> <Alert> <Diagnostics> <BEA-320016>
   <Creating diagnostic image in DOMAIN_HOME/servers
   /ProductManagementServer_1/adr/dia/ofm/SCMDomain/
   ProductManagementServer_1/incident/incdir_394 with a lockout minute
   period of 1.>

   The above message may not always appear after each stuck thread reported. It is
   printed at most four times an hour. If the message does not appear, manually look
   for the incident directory by checking the readme file in the subdirectories
   under the following directories:
   (UNIX) DOMAIN_HOME/servers/server_name/adr/dia/ofm/domain_name/server_
   name/incident
   (Windows) DOMAIN_HOME\servers\server_name\adr\dia\ofm\domain_name\server_
   name\incident

   The incident directory contains a WLDF diagnostic image which contains the JFR
   recording, and a file containing the thread dump

   For more information about diagnosing incidents, see the "Diagnosing Problems"
   chapter in the Oracle Application Server Administrator's Guide.
2. Review thread dump to see call stack of the thread. If thread is blocked waiting for lock, check what the thread holding the lock is doing.

3. If call stack involves executing JDBC calls, you can go to Grid Control and check the top activity around that time window, and see if there is a session with a matching module and action. See "Finding the Top SQL Queries" in the Oracle Fusion Applications Administrator's Guide.

4. Review the JRockit flight recording file JRockitFlightRecorder.jfr for more details. You will also need the ECID of the request which is recorded in the readme.txt file of the incident directory, and also the Oracle WebLogic Server log.

5. Perform the tasks in Section 2.2, "View Detailed Timing Of a Request Using a JRockit Flight Recorder (JFR) File."

Since the ECID of the request that caused the stuck thread is recorded in the error message, you also can follow the steps for troubleshooting slow requests that already have completed as documented in Section 2.1.3.1, "Troubleshooting Historical Requests."

### 2.1.3.4 Troubleshooting Slow Requests Using JFR Recording

See Section 2.2, "View Detailed Timing Of a Request Using a JRockit Flight Recorder (JFR) File."

### 2.1.3.5 Troubleshooting Memory Leaks and Heap Usage Pressure

#### Problem

Application performance degrades over time, heap usage and garbage collection activity increases overtime, sometimes OutOfMemoryErrors are seen. There could be memory leaks in the application, which causes the amount of free memory in the JVM to continuously decrease.

#### Solution

To solve this problem, perform the following:

1. Review the server_name.out file for OutOfMemoryErrors errors, which indicate a heap dump file has been written. The server_name.out file is located in the following directories:

   (UNIX) DOMAIN_HOME/servers/server_name/logs
   (Windows) DOMAIN_HOME\servers\server_name\logs

2. Restart the Managed Server.

   See the following documentation resources to learn more about other methods for starting and stopping the Managed Servers:

   - "Starting Managed Servers with a Startup Script" section in Oracle Fusion Middleware Managing Server Startup and Shutdown for Oracle WebLogic Server
   - "Starting Managed Servers with the java weblogic.Server Command" section in Oracle Fusion Middleware Managing Server Startup and Shutdown for Oracle WebLogic Server
   - "Starting and Stopping Managed Servers Using Fusion Middleware Control" section in Oracle Fusion Middleware Managing Server Startup and Shutdown for Oracle WebLogic Server
2.1.4 How to Troubleshoot Slow Batch Jobs

Batch jobs are often critical parts of key business processes. If throughput or execution time is beyond expectation, these steps may help diagnose the root cause.

2.1.4.1 Troubleshooting Jobs That Are in Wait/Ready/Blocked State for a Long Time

- Launch Oracle Enterprise Manager Cloud Control and look up the WebLogic Server domain target, or launch Fusion Applications Control for the domain.
- Expand Scheduling Service and select the one to which the job is submitted.
- Select Job Request > Search Job Request.
- Enter the appropriate search criteria.
- Expand the display to the request details. The top of the page will have an information block explaining why the job is not yet run.
- If many jobs are in a wait state, check Performance > Historical Reports > View: Requests Metrics By Work Assignment.
- Inspect Wait Time versus Processing Time.
  - If Wait Time is high, check several jobs and see why they are in a wait state.
  - If there is spare CPU capacity on the servers where the jobs are running, consider adding more threads to the work assignment.

2.1.4.2 Troubleshooting Jobs in Running State for a Long Time

See the Oracle Fusion Applications Administrator’s Guide.
Sometimes a job could be in a running state even if it had completed. See the ESS Troubleshooting Guide to first determine if job is still running.

Otherwise, open the request details page and click View Log Message to get the ECID. Go to the Java Diagnostics for the cluster to which the ESS job sends the request, such as for service calls, search for the ECID and see what, if any, Java Diagnostics data is recorded.

2.1.4.3 Troubleshooting Slow BI Publisher Jobs
See "Troubleshooting Oracle Enterprise Scheduler" in the Oracle Fusion Applications Administrator’s Guide.

Open the home page of the Oracle Fusion instance / product family / product in Cloud Control. The Top Long Running Job Requests and Recently Completed Job Requests region shows the health of ESS jobs. Since BI Publisher jobs are scheduled as ESS jobs, the health of those jobs will be shown in this region.

Click the request ID link to show the Request Detail page.

Select Actions > Request Log in the right hand side drop down and look for the log entries specific to this job request.

Select Actions > JVM Diagnostics in the right hand side drop down and view the health of this job request from the JVM perspective.

– The JVM page shows the data pertaining to the ECID corresponding to this job request.

– Inspect thread, cpu, memory and database diagnostic data for this specific job.

2.1.4.4 Troubleshooting Slow SOA Jobs

Open the home page of the Oracle Fusion product family in Cloud Control. The Recently Completed SOA Instances region shows the health of recently-completed SOA jobs in the product family context.

Click Composite Name to show the detailed information about the composite in the home page.

Click Dehydration Diagnostics to view the overall database activity associated with the dehydration store and any abnormal bottlenecks. To view database diagnostics, click individual SQL IDs.

Click Faults and Rejected Message to view more details about faulted SOA instances.

Select SOA Composites > Logs > View Log Messages to view the related log entries.

Select SOA Composites > Trace Instance to trace a particular SOA instance. To display all SOA instances, click Search without specifying a filter.

Click the instance id to view the complete trace details for this instance.

Click JVM Diagnostics to view the health of this SOA instance from a JVM perspective.

– The JVM page shows the data pertaining to the ECID corresponding to this job request.

– Inspect thread, cpu, memory and database diagnostic data for this specific job.
2.1.4.5 Troubleshooting Slow SQL Jobs

- From Cloud Control, look up the domain target, or go to Fusion Applications Control.
- Expand Scheduling Service and select the appropriate scheduling service group.
- On the right side, from the drop-down menu, select Job Requests > Search Job Requests and search for the job of interest.
- On the job details page, click the eye-glasses icon next to the Execution Type field. This will display a pop-up with the database session id that was used to process this job.
- From Enterprise Manager, follow these steps to get an ASH report for the session within the relevant time window to inspect top SQL statements and top wait events.
  - On the Performance page in the Average Active Sessions section, click Run ASH Report.
  - On the resulting Run ASH Report page, enter the date and time for the start and end of the time period for the report. (Presumably when some transient performance problem occurred.)
  - Click Generate Report and wait for the resulting ASH report that will appear under Report Results on the Run ASH Report page.
  - Click Save to File if you want to save the report in HTML format for future analysis.

2.1.4.6 Troubleshooting Slow Java Jobs

- Open the request details page and click View Log Message to get the ECID.
- Open the Java Diagnostics for the cluster to which the ESS job sends the request, such as for service calls, or the ESS server itself if most of the logic executes there.
- Search for the ECID to view any Java Diagnostics data that is recorded.

2.1.5 How to Troubleshoot a Slow BPEL Instance

- From Fusion Applications Control, select the SOA instance under SOA.
- Select Instances and search for the instance in question.
- Click the Instance link to display the flow trace to obtain the ECID.
- Click the BPEL component to see the audit trail that records the completion time for each step.
- Look for big gaps in the timing.
  - If a gap is due to a service call, use the request monitor and search for the ECID, check JVMD or JRockit Flight Recorder (JFR) to determine why it took so long.
2.1.5.1 Troubleshooting High Connection Usage

Problem
The connection usage on the Oracle Database is high, or there is an Oracle process on the database host consuming high amount of CPU.

Solution
To find out the source of the connection causing the high CPU on the database host and adjust the reference pool size from Fusion Applications Control:

1. Oracle Fusion Applications set values on a number of `v$session` attributes to indicate how the connection is being used. When looking at a connection consuming high CPU on the database, or when trying to understand what connections are used for what processes, inspect the value of these attributes as follows:

<table>
<thead>
<tr>
<th>Attribute in <code>v$session</code></th>
<th>Value Being Set</th>
</tr>
</thead>
<tbody>
<tr>
<td>Process</td>
<td>Data Source Name (for example, ApplicationDB)</td>
</tr>
<tr>
<td>Program</td>
<td>Oracle WebLogic Server Domain plus the Managed Server name, prefixed by DS (for example, DS/FinancialDomain/AccountsReceivableServer_1)</td>
</tr>
<tr>
<td>Module</td>
<td>Oracle Application Development Framework: ADF BC application module name</td>
</tr>
<tr>
<td></td>
<td>Oracle Enterprise Scheduler:</td>
</tr>
<tr>
<td></td>
<td>- Java job type: Class name, except oracle.apps</td>
</tr>
<tr>
<td></td>
<td>- PLSQL: the package and procedure name (for example, mypkg.myproc)</td>
</tr>
<tr>
<td></td>
<td>- Other jobs: Static: Executable name</td>
</tr>
<tr>
<td></td>
<td>Oracle BI Publisher: Name of the report</td>
</tr>
<tr>
<td>Action</td>
<td>Oracle Application Development Framework: jspx name</td>
</tr>
<tr>
<td></td>
<td>Oracle Enterprise Scheduler: Job definition name</td>
</tr>
<tr>
<td></td>
<td>Oracle BI Publisher: if request is submitted:</td>
</tr>
<tr>
<td></td>
<td>- Oracle Enterprise Scheduler: Oracle Enterprise Scheduler job definition name</td>
</tr>
<tr>
<td></td>
<td>- Oracle BI Publisher Scheduler Job: Oracle BI Publisher job name submitted by the user</td>
</tr>
<tr>
<td></td>
<td>- Oracle BI Publisher online: Static string (BIP:Online)</td>
</tr>
<tr>
<td></td>
<td>- Oracle BI Publisher Web services: Name of the web services</td>
</tr>
<tr>
<td>Client_Identifier</td>
<td>Application User Name</td>
</tr>
</tbody>
</table>

2. If the error messages related to connection pool capacity being reached are also seen in Oracle WebLogic Server logs, use the solution for connection leaks described in Section 2.1.5.2.
2.1.5.2 Troubleshooting Connection Leaks

Problem
When there are errors in the log, and the error message indicates connection pool size has been reached

Solution
To resolve this problem:

1. When data source is at maximum capacity and there are errors during connection reservation requests, then there may be connection leaks in the code

2. Enable JDBC profiling from the Oracle WebLogic Server Administration Console:
   a. In the Domain Structure, expand Services and then Data Sources.
   b. Click on the data source that needs to profiled, for example, ApplicationDB.
   c. In the Settings page, click on the Configuration tab, then click on Diagnostics subtab.
   d. Check the profiles that need to be collected (PROFILE_TYPE_CONN_USAGE_STR).
   e. Click Save.

3. Configure the diagnostic archive where the profiling data is saved from the Oracle WebLogic Server Administration Console:
   a. In the Domain Structure, expand Services, Diagnostics, and then Archives.
   b. Click on the server where you want to make changes (archives are stored for each server)
   c. In the Settings page, you can change archive location, size and how to retire data.
   d. Check the profiles that need to be collected (PROFILE_TYPE_CONN_USAGE_STR).
   e. Click Save.

4. To retrieve profiling data, use the sample code (http://download.oracle.com/docs/cd/E15051_01/wls/docs103/wl df_configuring/access_diag_data.html#wp1100898), with changes to the URL, username and password in the initialize method.

5. Run the sample code as a standalone program.

6. The program will capture the stack trace for each request for a connection from that data source. Inspect the callers to see the suspicious stack. This sample program requires connecting to a live Oracle WebLogic Server instance.

   The diagnostic archive file under the archive location can also be provided to Oracle Support for further analysis.

   Oracle WebLogic Server will not report a leak unless inactive connection timeout connection pool setting is set to a positive value. This cannot be done for Oracle Fusion Applications, as it will break functionality.
2.1.5.3 Troubleshooting Slow Requests Using SQL Trace
When a user reports that a specific operation is slow, and the slowness is reproducible and that slow database operations are suspected but the top activity reports did not provide sufficient information for resolving the problem.

To resolve this problem:
1. Enable SQL trace for the user session. See "Using SQL Tracing" in the Oracle Fusion Applications Administrator’s Guide.
2. Ask user to re-run the problematic flow and collect the SQL trace files and review

2.1.5.4 Troubleshooting Slow Oracle Enterprise Scheduler Jobs of SQL Type
When the user submits a SQL job type, the job remains in a RUNNING state for too long.

Solution
To resolve this problem, perform the following steps:
1. Use Fusion Applications Control to find the database session ID that was used to process the job:
   1. Search for the request, as described in "Searching for Oracle Enterprise Scheduler Job Requests" in the Oracle Fusion Applications Administrator’s Guide.
   2. On the Request Details page, in the Request Properties section, next to the Execution Type field, click the eye glasses icon.
      The Spawned Process Details dialog displays. This will bring up a pop-up with the database session id that was used to process this job
   3. Take note of the value in the Session Id field, and then click OK.
2. Use Grid Control to ensure the request processor and request dispatcher are running:
   a. Run an Active Session History (ASH) report for the session within the relevant time window to inspect top SQL statements and top wait events. See the "Resolving Transient Performance Problems" section in the Oracle Database 2 Day + Performance Tuning Guide.
   b. Identify time consuming SQL statements and tune following normal SQL tuning procedures. See "Finding the Top SQL Queries" in the Oracle Fusion Applications Administrator’s Guide.

2.1.5.5 Troubleshooting Excessive Activation

Problem
When response time suddenly increases with rising user count, even though there is no memory pressure, it is possible that the reference pool size for key application modules needs to be increased. If there is a JFR recording to review, and you observe many events containing callstacks containing the activateState method, you should also try adjusting the reference pool size.

Solution
To adjust the reference pool size from Fusion Applications Control:
1. Review the number of web sessions from Performance Summary pages:
For Fusion Applications Control:

a. From the navigation pane, expand the farm, Application Deployments.

b. From the Applications Deployments page, select the application.

c. From the Application Deployments menu, choose ADF > ADF Performance.

   The ADF Performance page displays.

d. Click the Application Module Pools tab.

e. Sort the request by descending order.

f. For the top 10 or so application modules, click the application module name to view the Activations count.

For Grid Control:

a. Click the Targets tab.

b. Click the Middleware secondary tab.

c. From the Search list, select Oracle WebLogic Server Domain, and then click Go.

d. Click on a domain.

   The WebLogic Server Domain home page displays.

e. In the table on the right-hand side of the page, expand the Application Deployments node.

f. Click the target application.

   The Application Deployment page displays.

g. From the Application Deployments menu, choose ADF > ADF Performance.

   The ADF Performance page displays.

h. In the Application Module Pools table, from the View list, select Total Requests, and once selected, from the Total Requests column, click Sort Descending.

i. For the top 10 or so application modules, click to see the details of each one.

j. After selecting an application module, on the Requests graph, from the Select metric to display in chart list, select Passivation and Activation to add to the graph.

   If activation count is close to passivation and is constantly above 0, then following Step 2 to adjust

2. If the activation count constantly increases, increase the application module reference pool size from Fusion Applications Control:

a. From the Application Deployments menu, choose ADF > Configure ADF Business Components.

   The ADF Configuration BC Configurations page displays.

b. From the Application Modules section, click the application module of interest. From the left hand side, select the local by selecting a name that ends in Local.

c. Click the Pooling and Scalability tab, and change the Reference Pool Size parameter.
2.1.6 How to Troubleshoot Oracle Business Intelligence

The usual indication that you should troubleshoot Oracle BI will be sluggish performance of BI components embedded across various applications. BI components can be in the form of such things as charts, tables, dashboards, and queries. Many of the configuration issues can be detected from the `nqquery.log` and `nqserver.log` log files.

A configuration problem generally will produce an error message. You would examine the same two log files for either sluggish response or an error.

Oracle BI Query Logging

To debug a query issue, you need to enable query logging. Query logging can be enabled by setting the LOGLEVEL variable to a value from 0 through 7. 0 denotes no logging, and 7 will generate detailed logging.

It is sufficient to use loglevel 2 to obtain logical BI server queries and the corresponding physical database queries in the log file.

Figure 2–1 shows how to use the Variable Manager to set the log level to 2.

**Figure 2–1 Setting the Log Level for Oracle BI**

Once you have enabled logs, you can obtain log files from `$ORACLE_INSTANCE/diagnostics/logs/OracleBIServerComponent/coreapplication_obis1/`

Relevant files are:

- `nqquery.log` - Will contain logical SQL, physical SQL and an execution plan chosen by OBIS for the logical SQL.
- `nqserver.log` - Will contain server-related data, such as initialization block-related errors.

Note that a single logical SQL can spawn multiple physical SQL statements.
If you run large queries and logs are being rotated quickly, you can modify parameters in `logconfig.xml` to control the frequency for log rotation. `logconfig.xml` is located at `$ORACLE_INSTANCE/config/OracleBIServerComponent/coreapplication_obis1/logconfig.xml` where `ORACLE_INSTANCE` is a path similar to `/u47/st99/instance/BIInstance`.

Once you have logging enabled, you can search for response time and identify where time is being spent by the query. In this example of a useful search string, logical query execution took 1 second, and physical query execution also took 1 second.

Physical query response time 1 (seconds)
Logical Query Summary Stats: Elapsed time 1, Response time 1, Compilation time 0 (seconds)

**BI Connection Pool Settings**

If you anticipate a higher load on a system, you can change the number of **Maximum connections** for various data sources to make resource use more efficient, as shown in Figure 2–2. This change must be propagated if the location of the Business Intelligence metadata is replaced.

**Figure 2–2 Changing the Maximum Number of Connections**

![Connection Pool Settings](image)

### 2.2 View Detailed Timing Of a Request Using a JRockit Flight Recorder (JFR) File

**Problem**
Certain requests are slow and there is a need to find out where time is spent

**Solution**
The JRockit Flight Recorder (JFR) file contains a record of various events that consume time, and can be used to help understand why a request is taking time.
To resolve this problem, create a JFR file:

1. Extract a JFR file from an Oracle WebLogic Server server by running the following command:

   (UNIX) `JROCKIT_HOME/bin/jcmd jrockit_pid dump_flightrecording recording=1 copy_to_file=path compress_copy=true`
   (Windows) `JROCKIT_HOME\bin\jcmd.exe jrockit_pid dump_flightrecording recording=1 copy_to_file=path compress_copy=true`

   See the "Running Diagnostic Commands" chapter in the *Oracle JRockit JDK Tools Guide* for more information about the *jrcmd* command-line tool.

2. To view the file, start the JRockit Mission Control Client from the following directories:

   (UNIX) `JAVA_HOME/bin/bin/jrmc`
   (Windows) `JAVA_HOME\bin\jrmc.exe`

3. Choose File > Open File to select the JFR file.

4. Locate the slowest requests or investigate a specific request:
To locate the slowest requests:

1. In the JRockitFlightRecorder.jfr page, click the Events icon.
2. Click the Log tab at the bottom of the page.
3. In the Event Type navigation pane on the left, locate Dynamic Monitoring System and then HttpRequest.
4. Click HTTP request; de-select all the other event types.
5. In the Log tab, in the Event Log section, click the Duration column to sort the duration in descending order.
   Each row corresponds to a HTTP Request and the duration column shows the response time for that request.
6. Click the row in the table to view the attributes of the requests.
7. In the Event Attributes sections, note the start time and the thread that serviced the request.

To investigate a specific request:

1. Find the Execution Context Identifier (ECID) of that request.
   If the request is related to an incident triggered by a STUCK thread, the incident readme.txt file will contain the ECID.
   Alternatively, you can search the Oracle WebLogic Server HTTP access.log for requests from specific users. See the “Viewing and Searching Log Files” section in the Oracle Application Server Administrator’s Guide.
2. In the JRockit Mission Control Client, in the JRockitFlightRecorder.jfr page, choose the WebLogic icon, and then
   If the Weblogic icon is not available, choose Help > Install Plugins to download the Oracle WebLogic Server plug-in.
3. Click the ECIDs tab at the bottom of the age.
4. In the ECIDs section, from Filter Column list, select ECID.
5. Enter the ECID in the search box and choose <Enter>.
6. In the results table, highlight the row with the matching ECID and right-click to bring up the menu.
7. Choose Operative Set > Clear, and then Operative Set > Add matching ECID > ECID to add the ECID to the operative set.
   This enables users to view only events associated with the operative set.
8. Click the Events icon.
9. In the Event Type navigation pane on the left, locate Dynamic Monitoring System and then HttpRequest.
10. Click HTTP request; de-select all the other event types. **
11. In the Event Log section, click Show Only Operative Set.
   Each row corresponds to the request with the matching ECID
12. Click the row in the table to view the attributes of the requests.
13. Note the start time and the thread that serviced the request.

5. Once the start time and the thread that serviced the request are identified, in the Logs tab, drag the time selector at the top of the screen to include only the time window for the duration of the request.
6. In the Event Log section, perform the following search:
   a. Deselect Show Only Operative Set.
   b. Enter the thread name in the search box.
   c. From the Filter Column list, select Thread.
d. Choose <Enter>.

7. In the **Event Type** navigation pane on the left, click the events of interest. Typically, these events are located under nodes **Dynamic Monitoring System**, **Java Application**, and **WebLogic > JDBC**.

The selected events appear in the table in the **Event Log** section.

8. Click the **Start Time** column to sort by the time when these events occur, or click the **Duration** column to view the events that took longest.

The **JDBC Statement Execute** events correspond to SQL execution. If there are slow SQL statements, the event details give the SQL text. These events do not have callstacks.

9. To see the callstack for slow SQL statements, view the **Socket Read** event that happens right after the **JDBC Statement Execute** event.

This event corresponds to Oracle WebLogic Server waiting for the SQL results to return, and it has callstack in the event details.

10. Review the callstacks for long **Java Blocked** and **Java Wait** events to see if the cause can be identified. See the "Analyzing Flight Recorder Data in JRockit Mission Control" section in the Oracle Fusion Middleware Configuring and Using the Diagnostics Framework for Oracle WebLogic Server.

11. If more details are needed to compare with what is captured in the default recording, and the user can reproduce the slowness, start an explicit recording. See the "Starting an Explicit Recording" section in the Oracle JRockit Flight Recorder Run Time Guide.

### 2.3 Using My Oracle Support for Additional Troubleshooting Information

You can use My Oracle Support (formerly MetaLink) to help resolve Oracle Fusion Applications problems. My Oracle Support contains several useful troubleshooting resources, such as:

- Knowledge base articles
- Community forums and discussions
- Patches and upgrades
- Certification information

**Note:** You can also use My Oracle Support to log a service request.

You can access My Oracle Support at [https://support.oracle.com](https://support.oracle.com).
This chapter discusses database tweaks that are specific to Oracle Fusion Applications. This chapter contains the following sections:

- Section 3.1, "Introduction"
- Section 3.2, "Tuning the Database"

### 3.1 Introduction

Oracle Fusion Applications set values on a number of v$session attributes to indicate how the connection is being used. When looking at a connection consuming high CPU on the database, or when trying to understand what connections are used for what processes, these attribute values may help provide the answer.

<table>
<thead>
<tr>
<th>Attribute in v$session</th>
<th>Value Being Set</th>
</tr>
</thead>
<tbody>
<tr>
<td>Process</td>
<td>Data Source Name (such as ApplicationDB)</td>
</tr>
<tr>
<td>Program</td>
<td>WebLogic Server Domain plus Managed Server name, prefixed by DS (such as DS/FinancialDomain/AccountsReceivableServer_1)</td>
</tr>
<tr>
<td>Module</td>
<td>For ADF: ADF BC application module name</td>
</tr>
<tr>
<td></td>
<td>For ESS:</td>
</tr>
<tr>
<td></td>
<td>For Java job type, the class name (minus oracle.apps)</td>
</tr>
<tr>
<td></td>
<td>For PL/SQL, the package and procedure name (such as mypkg.myproc)</td>
</tr>
<tr>
<td></td>
<td>For other job types, the executable name should be passed</td>
</tr>
<tr>
<td></td>
<td>For BI Publisher: the name of the report</td>
</tr>
</tbody>
</table>
3.2 Tuning the Database

Table 3–2 provides the database initialization parameter guidelines for Oracle Fusion Applications. Within the Oracle Fusion Applications ecosystem, there exist four types of databases. They are Online Transaction Processing (OLTP) Starter or Production configuration, IDM, and DW. These parameter values are intended to provide a baseline. The database that is installed during Oracle Fusion Applications provisioning is configured with the suggested values. As your deployment and workloads characteristics change, these values may need to be adjusted.

Tuning the database involves adjusting the sizing parameters based on the available resource and load on the database. The sga_target, pga_aggregate_target and processes parameters from Table 3–2 are examples of such parameters that need to be tuned based on SGA, PGA advisories and looking into the number of open processes during peak load.

In addition, you may consider setting a minimum value for SHARED_POOL_SIZE and DB_CACHE_SIZE to minimize frequent resizing.

Table 3–2 Common init.ora Parameters

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Description, Default value and suggested start value</th>
</tr>
</thead>
<tbody>
<tr>
<td>audit_trail</td>
<td>Enables or disables database auditing. Oracle Release 2 (11.2.0.2) default value: DB. The suggested value for all Oracle Fusion Applications databases is NONE.</td>
</tr>
<tr>
<td>_fix_control</td>
<td>This parameter addresses the dynamic sampling of global temporary tables. Oracle Release 2 (11.2.0.2) default value: N/A. The suggested value for an Oracle Fusion Applications Starter or Production database is 6708183:ON</td>
</tr>
</tbody>
</table>
### Table 3–2 (Cont.) Common init.ora Parameters

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Description, Default value and suggested start value</th>
</tr>
</thead>
<tbody>
<tr>
<td>plssql_code_type</td>
<td>PLSQL_CODE_TYPE specifies the compilation mode for PL/SQL library units.</td>
</tr>
<tr>
<td></td>
<td>INTERPRETED: PL/SQL library units are compiled to PL/SQL bytecode format. Such modules are executed by the PL/SQL interpreter engine.</td>
</tr>
<tr>
<td></td>
<td>NATIVE: PL/SQL library units are compiled to native (machine) code. Such modules are executed natively without incurring any interpreter impacts.</td>
</tr>
<tr>
<td></td>
<td>Oracle Release 2 (11.2.0.2) default value is INTERPRETED. The suggested value for all Oracle Fusion Applications databases is NATIVE.</td>
</tr>
<tr>
<td>nls_sort</td>
<td>NLS_SORT specifies the collating sequence for ORDER BY queries.</td>
</tr>
<tr>
<td></td>
<td>■ If the value is set to BINARY, the collating sequence for ORDER BY queries is based on the numeric value of characters (a binary sort that requires fewer system resources).</td>
</tr>
<tr>
<td></td>
<td>■ If the value is a named linguistic sort, sorting is based on the order of the defined linguistic sort. Most (but not all) languages supported by the NLS_LANGUAGE parameter also support a linguistic sort with the same name.</td>
</tr>
<tr>
<td></td>
<td>Oracle Release 2 (11.2.0.2) default value: Derived from NLS_LANGUAGE. The suggested value for all Oracle Fusion Applications databases is BINARY.</td>
</tr>
<tr>
<td>open_cursors</td>
<td>Specifies the maximum number of open cursors (handles to private SQL areas) a session can have at once. It is important to set the value of OPEN_CURSORS high enough to prevent your application from running out of open cursors.</td>
</tr>
<tr>
<td></td>
<td>Oracle Release 2 (11.2.0.2) default value is 50. The suggested value for all Oracle Fusion Applications databases is 500.</td>
</tr>
<tr>
<td>session_cached_cursors</td>
<td>Specifies the number of session cursors to cache. Repeated parse calls of the same SQL statement cause the session cursor for that statement to be moved into the session cursor cache. Subsequent parse calls find the cursor in the cache and do not reopen the cursor. Oracle uses a least recently used algorithm to remove entries in the session cursor cache to make room for new entries when needed. This parameter also constrains the size of the PL/SQL cursor cache that PL/SQL uses to avoid having to re-parse as statements are re-executed by a user.</td>
</tr>
<tr>
<td></td>
<td>Oracle Release 2 (11.2.0.2) default value is 50. The suggested value for all Oracle Fusion Applications databases is 500.</td>
</tr>
<tr>
<td>_b_tree_bitmap_plans</td>
<td>This enables use of bitmap access paths for b-tree indexes.</td>
</tr>
<tr>
<td></td>
<td>Oracle Release 2 (11.2.0.2) default value: TRUE The suggested value for all Oracle Fusion Applications databases is FALSE.</td>
</tr>
<tr>
<td>processes</td>
<td>Sets the maximum number of operating system processes that can be connected to Oracle concurrently. The value of this parameter must account for Oracle background processes. SESSIONS parameter is deduced from this value.</td>
</tr>
<tr>
<td></td>
<td>Oracle Release 2 (11.2.0.2) default value: 100 The suggested value for an Oracle Fusion Applications DW or IDM database (OID, OIM) is 2500. The suggested value for an Oracle Fusion Applications Starter or Production database is 5000.</td>
</tr>
<tr>
<td>sga_target</td>
<td>Setting this parameter to a nonzero value enables Automatic Shared Memory Management. Consider using automatic memory management, both to simplify configuration and to improve performance.</td>
</tr>
<tr>
<td></td>
<td>Oracle Release 2 (11.2.0.2) default value: 0 The suggested value for Oracle Fusion Applications IDM databases (OID, OIM) is 4 GB. The suggested value for an Oracle Fusion Applications Starter database is 9 GB. The suggested value for an Oracle Fusion Applications DW database is 8 GB. The suggested value for an Oracle Fusion Applications Production database is 18 GB (based on reference hardware having 32 GB of physical memory).</td>
</tr>
</tbody>
</table>
This section contains an overview for configuring a database for performance. Although it is expected that modifications will be made to the database on an ongoing basis to maintain or improve performance, significant benefits can be gained by proper initial configuration of the database.

In addition to tuning the database parameters, the database administrator should properly configure the REDO Logs, and the UNDO and TEMP tablespaces, to meet the demands of the expected or observed database workload. This is an empirical task. The recommendations in this section are intended to provide initial guidance in these areas.

### UNDO

It is recommended that the default mode of automatic undo management be leveraged to maximize performance and efficiency. Oracle Enterprise Manager Fusion Applications Control Automatic Undo Management Advisor should be leveraged to set configuration details for undo tablespace and retention settings. This advisor also provides access to the Undo Advisor that assesses the effect and provides advice of a new undo retention setting. The suggested minimum size for the UNDO tablespace is 6 GB with auto-extend enabled.

### TABLESPACES

It is recommended that the location of the datafiles be optimized for I/O performance and growth. It is also recommended that Fusion Applications Control Segment Advisor be leveraged to optimize the use of segment space and assure performance degradation does not occur. The advisor can provide historical growth trends of segments, which can be used to proactively plan for growth.
**TEMP**

Oracle recommends the use of locally-managed temporary tablespaces with UNIFORM extents and the default size of 1 MB. Some workflows in Oracle Fusion applications can generate a large amount of disk sorts that require high temporary space requirements. Therefore, the use of multiple temporary tablespaces and tablespace groups is recommended to meet these requirements and assure optimal performance. A suggested minimum size for the TEMP tablespace (or tablespace group) assigned to the Oracle Fusion Applications schema owners is 6 GB with auto-extend enabled.

**REDO LOGS**

Under demanding workloads, the size of the redo log files can influence performance. Generally, larger redo log files provide better performance. Undersized log files increase checkpoint activity and reduce performance. You can obtain sizing advice on the Redo Log Groups page of Fusion Applications Control. In addition, depending on your storage configuration and performance characteristics, you may need to redistribute redo logs to optimize I/O performance. The suggested minimum setting for Redo Logs is to have 3 log files of 2 GB each.

### 3.2.2 How to Configure Kernel Parameters

The parameters listed here are only a subset of the parameters needed to properly configure your database environment. Consult the Oracle Database Installation Guide for your specific operating system to determine the overall requirements for the kernel parameter settings. Installation guides are available in the Oracle Database Documentation Library.

If the current values of these parameters are larger than what is recommended, no change is necessary.

- **SHMMAX** - Set to the larger of either the largest System Global Area (SGA) on the system or half of the physical memory available.
- **SEMMNS** - Set to the larger of either 32000 or twice the sum of the PROCESSES initialization parameter for each Oracle database.
- **SHMALL** - Set to the sum of all SGAs divided by the page size (getconf PAGE_SIZE).
- **SHMMNI** - Recommended value is 4096.
- **vm.nr_hugepages** - Should be set for SGA larger than 8 GB. The value should be set to the sum of all SGAs.

### 3.2.3 How to Configure the Database Listener

The database listener is responsible for accepting and routing connection requests to the database. The parameters described in this section can change the behavior of the listener and, therefore, need to be examined and changed accordingly to address the needs of your particular database connection load.

**SQLNET.EXPIRE_TIME Parameter**

For Fusion application environments that leverage a firewall between the middle and database tiers, it is recommended to set the database listener setting SQLNET.EXPIRE_TIME. This parameter is defined in the sqlnet.ora file that is located in the TNS_ADMIN location. This parameter is used to set a time interval, in minutes, to determine how often to probe connections to verify their status.
Typically, a firewall is configured to terminate connections after it has been idle for a specified amount of time. Setting SQLNET.EXPIRE_TIME to an interval smaller than the firewall’s will generate enough traffic within an appropriate interval to prevent the firewall from determining the connections are in an idle state and terminating these connections. A setting of SQLNET.EXPIRE_TIME = 10 is commonly used.

**INBOUND_CONNECT_TIMEOUT Parameter**

On occasion, due to network latency, connections can exceed the default timeout of 60 seconds. Typical symptoms of this include observing "TNS-12537: TNS:connection closed" errors in the database alert logs. One of the things that can resolve these issues is to increase the timeout parameters. A suggested value of INBOUND_CONNECT_TIMEOUT_listenerName = 120 (listener.ora) and SQLNET.INBOUND_CONNECT_TIMEOUT=130 (sqlnet.ora) is recommended.

3.2.4 How to Tune the Real Application Cluster (RAC)

Oracle Real Application Clusters (RAC) is a cluster database with a shared cache architecture that overcomes the limitations of a traditional shared-nothing approach to provide highly scalable and available database solutions for all business applications.

RAC supports the transparent deployment of a single database across a cluster of servers, providing fault tolerance from hardware failures or planned outages in terms of availability, scalability, and low-cost computing.

RAC provides:

- **High Availability**
  
  RAC provides the highest availability for applications by removing the single server as a single point of failure.

- **Flexible Scalability**
  
  When more processing power and resources are needed, adding a server to the database cluster without taking users offline will gain horizontal scalability.

- **Automatic Workload Management**
  
  Application workloads can be individually managed and controlled using managed services. Users connecting to a service are load balanced across the server pool.

**Initial Setting**

The network between the nodes of a RAC cluster must be private.

Supported links: Giga-bit Ethernet and InfiniBand

Supported transport protocols: UDP or RDS

Use multiple or dual-ported Network Interface Cards (NICs) for redundancy and increase bandwidth with NIC bonding.

Set these parameters with respect to the ORACLE/OS release for Interconnect performance.

- `net.core.rmem_default = 262144`
- `net.core.rmem_max = 262144`
- `net.core.wmem_default = 262144`
- `net.core.wmem_max = 262144`
**Troubleshooting**

Use this information to identify if proper interconnect is used and picked up by clusterware.

- To identify the interconnect:
  
  ```
  show parameter cluster_interconnects
  or
  select name,ip_address,is_public from v$cluster_interconnects
  ```

  The Alert log will have an entry similar to this:

  Cluster communication is configured to use the following interface(s) for this instance
  123.45.67.89
  cluster interconnect IPC version:Oracle UDP/IP (generic)

- Identify network and contention issues.

  Check for "gc cr lost blocks" wait event in Automatic Workload Repository (AWR)/sysstats.

  If found, check for these errors on the NIC:
  - Dropped packets/fragments
  - Buffer overflows
  - Packet reassembly failures or timeouts
  - TX/RX errors

  Use these commands to find any errors:

  ```
  netstat -s
  ifconfig -a
  ORADEBUG
  ```

- Identify Interconnect performance from AWR.

  - Under **Global Cache and Enqueue Services - Workload Characteristics**
    
    Avg global cache cr block receive time (ms): should be <=15 ms

  - **Global Cache and Enqueue Services - Messaging Statistics**
    
    Avg message sent queue time on ksxp (ms): should be <1 ms

  - **Under Interconnect Ping Latency Stats**
    
    Avg Latency 8K msg should be close to Avg Latency 500B msg.

- These wait events from AWR/sysstat can indicate contention related to RAC.

  GC current block busy
  GV cr block busy
  GC current buffer busy
  GC buffer busy acquire/release

  See the *Oracle Real Application Clusters Administration and Deployment Guide* for a complete list.

  These wait events in the AWR indicate that there might be a Hot Block that is causing these wait events. From the AWR Segment Statistics, you can find the objects.

  `Enq:TX Index Contention`
Gc buffer busy
Gc current block busy
Gc current split

This issue will be noticed if multiple sessions are inserting into a single object or are using a sequence, and the indexed column is sequentially increasing. To address the specific issues:

– Identify the indexes and Global Hash Partition them.
– Increase the Sequence Cache if ordering is not a problem.

### 3.2.5 How to Optimize SQL Statements

Oracle Fusion Applications use Cost Based Optimization (CBO) to choose the most efficient execution plan for SQL statements. Using this approach, the optimizer determines the most optimal execution plan by costing available access paths and factoring information based on statistics for the schema objects accessed by the SQL statement.

#### 3.2.5.1 Collecting Optimizer Statistics

For the query optimizer to produce an optimal execution plan, the statistics in the data dictionary should accurately reflect the volume and data distribution of the tables and indexes. To this end, database statistics should be refreshed periodically. However, that does not necessarily imply that you should gather statistics frequently. Systems that are close to going live typically experience inserts of a large amount of data, as data from legacy systems is migrated. In that scenario, the statistics would probably need to be refreshed quite frequently (for instance, after each major load), as large loads could change the data distribution significantly. Once the system reaches steady state, the frequency of statistics collection at the schema/database level should be reduced to something like once a month. However, statistics on some volatile tables can be gathered as frequently as required.

Oracle Fusion Applications has an automated way of gathering the statistics. It uses DBMS_STATS with the AutoTask feature. For more information about AutoTask, see "Enabling and Disabling Automatic Optimizer Statistics Collection" in the Oracle Database Performance Tuning Guide.

**Gathering Statistics for Cost Based Optimizer**

Oracle Fusion applications use automatic optimizer statistics collection. That is, the database automatically gathers the optimizer statistics. Automatic statistics eliminates any manual intervention, thereby significantly reducing the onus on a system administrator. For a DW database, optimizer statistics are collected as part of ETL and BI apps, so it is recommended not to use automatic statistics gathering.

**Table Statistics and Number of Distinct Values (NDV)**

For versions prior to Oracle 12g, a database performance problem can occur by missing a non-popular value in the sample created from the table, leading to a frequency histogram where the number of buckets is less than the number of values.

To work around this problem, set this entry in init.ora:

```
_fix_control='5483301:off','6708183:ON';
```
Manual Statistics Gathering

Automatic optimizer statistics collection is sufficient for most of the database objects but in a Fusion database that are close to going live or for tables that are modified significantly, manual statistic gathering is needed.

In these cases, use the DBMS_STATS.GATHER_TABLE_STATS procedure. Do not explicitly set any parameters apart from owner and table_name.

All the parameters, such as estimate_percent, parallel degree and method_opt, are globally seeded for Oracle Fusion database tables. Therefore, there is no need to use them when manually gathering the statistics. See Figure 3–1.

Figure 3–1  Using the Fusion Applications Control to Gather the Statistics

Histograms

All the columns that need a histogram are seeded using DBMS_STATS.SET_TABLE_PREFS and the automatic optimizer statistic collection will create histograms for all the pre-identified columns apart from the columns identified by DBMS_STATS under size auto.

You can use DBMS_STATS.SET_TABLE_PREFS to manually seed histograms. See Figure 3–2

Figure 3–2  Using Oracle Enterprise Manager Grid Control

MDS DB: Collecting database statistics for optimizing the MDS database repository performance

Ensure auto-stats collection is enabled.
In most cases, the first 32 characters of PATH_FULLNAME in the MDS_PATHS table are the same. You can prevent the database putting them in same bucket of histogram by doing the following:

- Drop the histogram for PATH_FULLNAME column by executing the following as system.
  
  ```sql
  execute dbms_stats.delete_column_stats(ownname=>'mdsSchemaOwner',
  tabname=>'MDS_PATHS', colname=>'PATH_FULLNAME', col_stat_type=> 'HISTOGRAM');
  ```

- Set table preferences to exclude collecting histogram for PATH_FULLNAME column.
  
  ```sql
  execute dbms_stats.set_table_prefs(mdsSchemaOwner, 'MDS_PATHS', 'METHOD_OPT',
  'FOR_COLUMNS SIZE 1 PATH_FULLNAME');
  ```

### 3.2.5.2 Pinning Packages and Cursors

Pinning the objects in the shared pool reduces the possibility of ora-4031 error messages, and increases the performance of the OLTP applications.

Objects are cached in a library cache that uses a Least Recently Used (LRU) algorithm to flush the objects. The problem worsens if these large library cache objects are executed only infrequently. That is, if they are loaded into the library cache whenever required if they have aged out of the cache. This causes most of the problems leading to ora-4031 errors and poor performance for UI-specific flows.

Consider a large package, or any object, that has to be loaded into the shared pool. Large PL/SQL objects present particular challenges. The database has to search for free space for the object. If it cannot get enough contiguous space, it will free many small objects to satisfy the request. If several large objects need to be loaded, the database has to throw out many small objects in the shared pool. Finding candidate objects and freeing memory is very costly. These tasks will affect CPU resources.

The same situation applies to SQL statements that are executed occasionally but are very important from a response time perspective. For example, a statement that is part of a CEO's dashboard is executed once every 3 to 4 hours, but it takes a long time for parsing.

A common question asked is how to keep the LRU algorithm from forcing objects out of the shared pool. This is mainly useful in cases where a SQL statement that is very expensive to parse may just be executed once every 12 hours, such as a dashboard SQL statement. But due to the LRU algorithm, the results are flushed before the next execution. In these cases, you may prefer to pin the cursor so that subsequent parse times are reduced.

The DBMS_SHARED_POOL package provides procedures to facilitate this. With this, you have an easy way to ensure that the specified cursors always remain in the Most Recently Used (MRU) end of the cache. This prevents the cursor from being paged out and then re-parsed upon re-load. The DBMS_SHARED_POOL.KEEP procedure is used to pin the cursor and DBMS_SHARED_POOL.UNKEEP is used to unpin the cursor.

### 3.2.6 How to Configure the Database Resource Manager

To better ensure system stability during periods of high system load and prevent runaway queries, Oracle Database Resource Manager can be enabled on the Oracle Fusion Applications database instance.
For more details about the feature, including an explanation on when enabling this feature is desired, see the Oracle Database Administrator’s Guide.

Use the resource manager to:

- Assign connections to different resource consumer groups.
- Create directives to manage resource allocations for connections in different resource consumer groups.

### 3.2.6.1 Assigning Connections to Resource Consumer Groups

There are multiple ways a connection can be assigned to a resource consumer group. See the Oracle Database Administrator’s Guide for complete details.

See Table 3-1 for connection attributes settings for Oracle Fusion Applications.

### 3.2.6.2 Using the Oracle Database Resource Manager

This section shows how to use the Oracle Database Resource Manager for Oracle Fusion Applications. Connection attribute mappings, and explicit assignment using a database login trigger are used to assign connections to different resource consumer groups. The sample resource plan is called FUSION SAMPLE PLAN.

The sample plan has two resource consumer groups:

- **FUSION SAMPLE ONLINE GROUP**: Connections used for servicing ADF UI pages are assigned to this resource group via a login trigger.
- **FUSION SAMPLE BATCH GROUP**: All other connections used by Fusion Applications are assigned to this group.

These resource directives are defined:

- Connections in FUSION SAMPLE ONLINE GROUP get priority for 45% of the CPU. Parallel query is disabled, and any queries consuming more than 120s of CPU or resulting in more than 10GB of I/O will be canceled.
- Connections in FUSION SAMPLE BATCH GROUP get priority for 35% of the CPU. There are no other resource restrictions.
- Connections used for sysdba activities get priority for 15% of the CPU. There are no other resource restrictions.
- All non-Fusion Applications connections get priority for 5% of the CPU.

Note that if a particular resource group does not consume all its allocated CPU, the unused CPU can be used by other resource consumer groups.

To map connections to resource consumer groups, this approach is used:

- All schemas containing the string FUSION are mapped to FUSION SAMPLE BATCH GROUP.
- A login trigger is used to check connections to schema FUSION_RUNTIME. If the connection is coming from an ADF server, assign the connection to FUSION SAMPLE ONLINE GROUP.

To enable this sample resource plan, follow these steps:

- Run this script using sqlplus as sysdba to create the resource plan, the resource consumer groups and the resource directives.

```sql
begin
    DBMS_RESOURCE_MANAGER.CLEAR_PENDING_AREA();
    DBMS_RESOURCE_MANAGER.CREATE_PENDING_AREA();
end
```
begin
    dbms_resource_manager.delete_plan('FUSION_SAMPLE_PLAN');
exception
    when others then
        null;
end;

begin
    dbms_resource_manager.delete_consumer_group(CONSUMER_GROUP => 'FUSION_SAMPLE_ONLINE_GROUP');
exception
    when others then
        null;
end;

begin
    dbms_resource_manager.delete_consumer_group(CONSUMER_GROUP => 'FUSION_SAMPLE_BATCH_GROUP');
exception
    when others then
        null;
end;

dbms_resource_manager.create_consumer_group(CONSUMER_GROUP =>'FUSION_SAMPLE_ONLINE_GROUP', COMMENT => 'Consumer Group for online users');
dbms_resource_manager.create_consumer_group(CONSUMER_GROUP =>'FUSION_SAMPLE_BATCH_GROUP', COMMENT => 'Consumer Group for batch');
dbms_resource_manager.create_plan(PLAN => 'FUSION_SAMPLE_PLAN', COMMENT => 'Fusion Applications Resource Plan');
dbms_resource_manager.create_plan_directive(
    plan => 'FUSION_SAMPLE_PLAN',
    group_or_subplan => 'FUSION_SAMPLE_ONLINE_GROUP',
    comment => 'Online users at level 1',
    mgmt_p1 => 45,
    parallel_degree_limit_p1 => 0,
    switch_time => 120,
    switch_io_megabytes => 10000,
    switch_group => 'CANCEL_SQL');
dbms_resource_manager.create_plan_directive(
    plan => 'FUSION_SAMPLE_PLAN',
    group_or_subplan => 'FUSION_SAMPLE_BATCH_GROUP',
    comment => 'Batch users at level 1',
    mgmt_p1 => 35);
dbms_resource_manager.create_plan_directive(
    plan => 'FUSION_SAMPLE_PLAN',
    group_or_subplan => 'SYS_GROUP',
    comment => 'System administrator group at level 1',
    mgmt_p1 => 15);
dbms_resource_manager.create_plan_directive(
    plan => 'FUSION_SAMPLE_PLAN',
    group_or_subplan => 'OTHER_GROUPS',
    comment => 'Other users at level 1',
    mgmt_p1 => 5);
for rec in (select username from dba_users where username like '%FUSION%')
loop
    if (rec.username <> 'FUSION_READ_ONLY') then
        dbms_resource_manager.set_consumer_group_mapping('ORACLE_USER', rec.username, 'FUSION_SAMPLE_BATCH_GROUP');
    end if;
end loop;

begin
    dbms_resource_manager.set_consumer_group_mapping('ORACLE_USER', 'SEARCHSYS', 'FUSION_SAMPLE_BATCH_GROUP');
exception
    when others then
        null;
end;
DBMS_RESOURCE_MANAGER.VALIDATE_PENDING_AREA();
DBMS_RESOURCE_MANAGER.SUBMIT_PENDING_AREA();
DBMS_RESOURCE_MANAGER.CLEAR_PENDING_AREA();
dbms_resource_manager_privs.grant_switch_consumer_group('PUBLIC', 'FUSION_SAMPLE_ONLINE_GROUP', FALSE);
dbms_resource_manager_privs.grant_switch_consumer_group('PUBLIC', 'FUSION_SAMPLE_BATCH_GROUP', FALSE);
end;
/

As sysdba, issue the following command to enable the resource plan:

(if using spfile)
ALTER SYSTEM SET RESOURCE_MANAGER_PLAN = FUSION_SAMPLE_PLAN SCOPE = BOTH;

(if not using spfile)
ALTER SYSTEM SET RESOURCE_MANAGER_PLAN = FUSION_SAMPLE_PLAN;

Run this script using sqlplus as sysdba to create the login trigger:

CREATE OR REPLACE TRIGGER fusion_resource_trigger
    AFTER logon ON fusion_runtime.schema
declare
    login_sid pls_integer;
    login_program varchar2(40);
    old_plan21 varchar2(44);
begin
    select 'Y' into login_program from dual where exists
        (select program from v$session where audsid=userenv('sessionid') and
            process like 'ApplicationDB%' and
            ( program is not null and
                program not like 'JDBC Thin Client' and
                program not like '%ess_server%' and
                program not like '%soa_server%' and
                program not like '%SearchServer%' and
                program not like '%search_server%' and
                program not like '%odi_server%' and
                program not like '%bi_server%' ) );
    if login_program = 'Y'
        then
            dbms_session.switch_current_consumer_group('FUSION_SAMPLE_ONLINE_GROUP', old_plan21, false);
    end if;
end;
else
null;
end if;
EXCEPTION
when NO_DATA_FOUND
then
null;
end;
/

- If applicable, change the init.ora parameter to enable the resource plan:

```
RESOURCE_MANAGER_PLAN = FUSION_SAMPLE_PLAN
```

- Bounce all middle tiers.

### 3.2.6.3 Monitoring the Resource Manager

For details, see the *Oracle Database Administrator’s Guide*. In particular, `v$rsrcc_consumergroups` provides information on CPU wait time and the number of SQL statements canceled.

Scripts and Tips for Monitoring CPU Resource Manager, note ID 1338988.1 in My Oracle Support, also has information about monitoring CPU usage for different resource consumer groups. If a resource consumer group is spending significant time waiting for CPU, review the resource directives and see if the waits are expected and if any adjustments are needed.